

Verificação comportamental usando o ModelSim

Prof. Dr. Oscar Eduardo Anacona Mosquera

oscar.mosquera@ufmt.br

19 de março de 2024

Objetivos

RTL viewer

Testbench

Configuração do Quartus

ModelSim

Zoom do waveform

Sinais do waveform

Configuração do arquivo RTL

Conteúdo

- 1 **Objetivos**
- 2 **RTL viewer**
- 3 **Testbench**
 - Configuração do Quartus
- 4 **ModelSim**
 - Zoom do waveform
 - Sinais do waveform
 - Configuração do arquivo RTL

● RTL no Quartus:

- ▶ Entender os conceitos e princípios do RTL (Register-Transfer Level) e como ele é aplicado no software Quartus II.
- ▶ Aprender a utilizar o Quartus II para realizar a descrição em RTL de circuitos digitais.
- ▶ Explorar as ferramentas e recursos disponíveis no Quartus para desenvolver designs em RTL e realizar simulações para verificação.

● Testbench:

- ▶ Definir o conceito de testbench e sua importância no processo de verificação de circuitos digitais.
- ▶ Configurar e utilizar testbenches no Quartus II para simular designs em VHDL.
- ▶ Construir testbenches eficazes para validar o comportamento dos circuitos implementados no FPGA DE0 Nano.

● ModelSim:

- ▶ Introdução ao ModelSim como uma ferramenta de simulação de circuitos digitais.
- ▶ Explorar as funcionalidades do ModelSim para realizar simulações precisas e eficientes de designs VHDL.
- ▶ Aprender a interpretar os resultados das simulações no ModelSim para verificar o funcionamento correto dos circuitos.

Conteúdo

- 1 **Objetivos**
- 2 **RTL viewer**
- 3 **Testbench**
 - Configuração do Quartus
- 4 **ModelSim**
 - Zoom do waveform
 - Sinais do waveform
 - Configuração do arquivo RTL

Objetivos

RTL viewer

Testbench

Configuração do Quartus

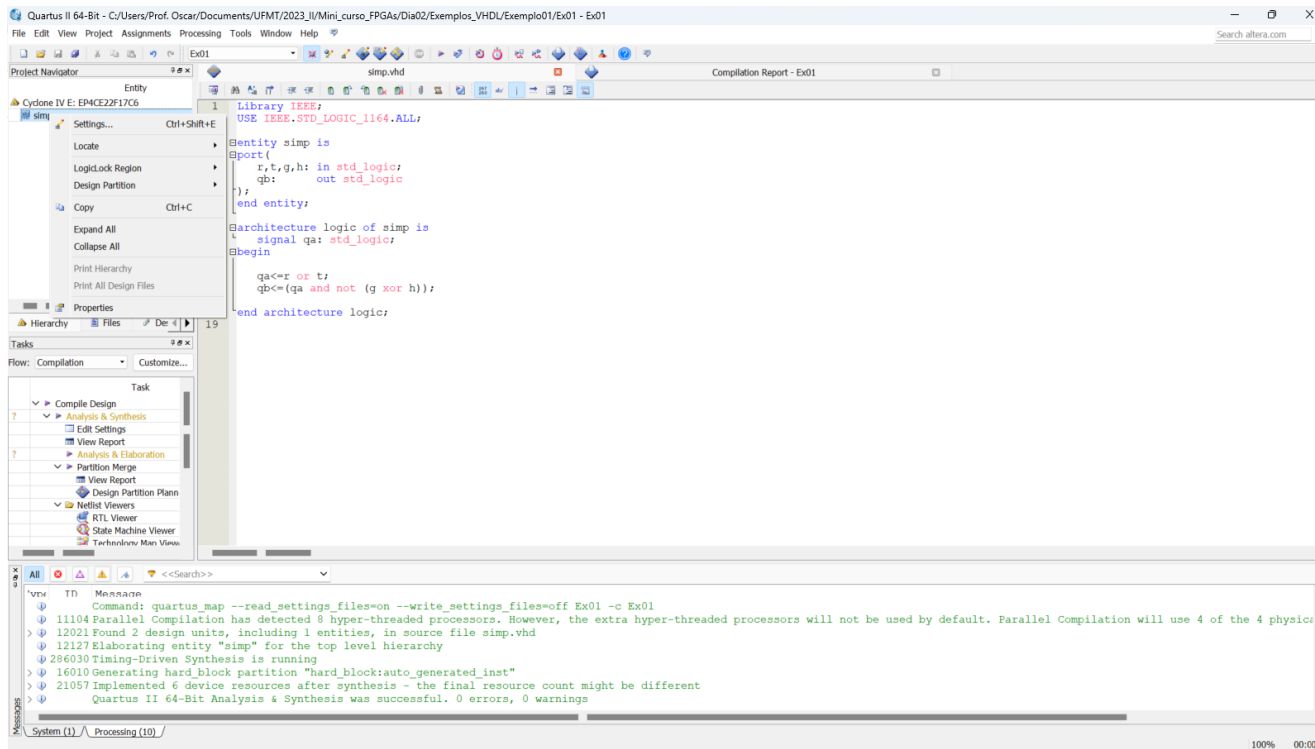
ModelSim

Zoom do waveform

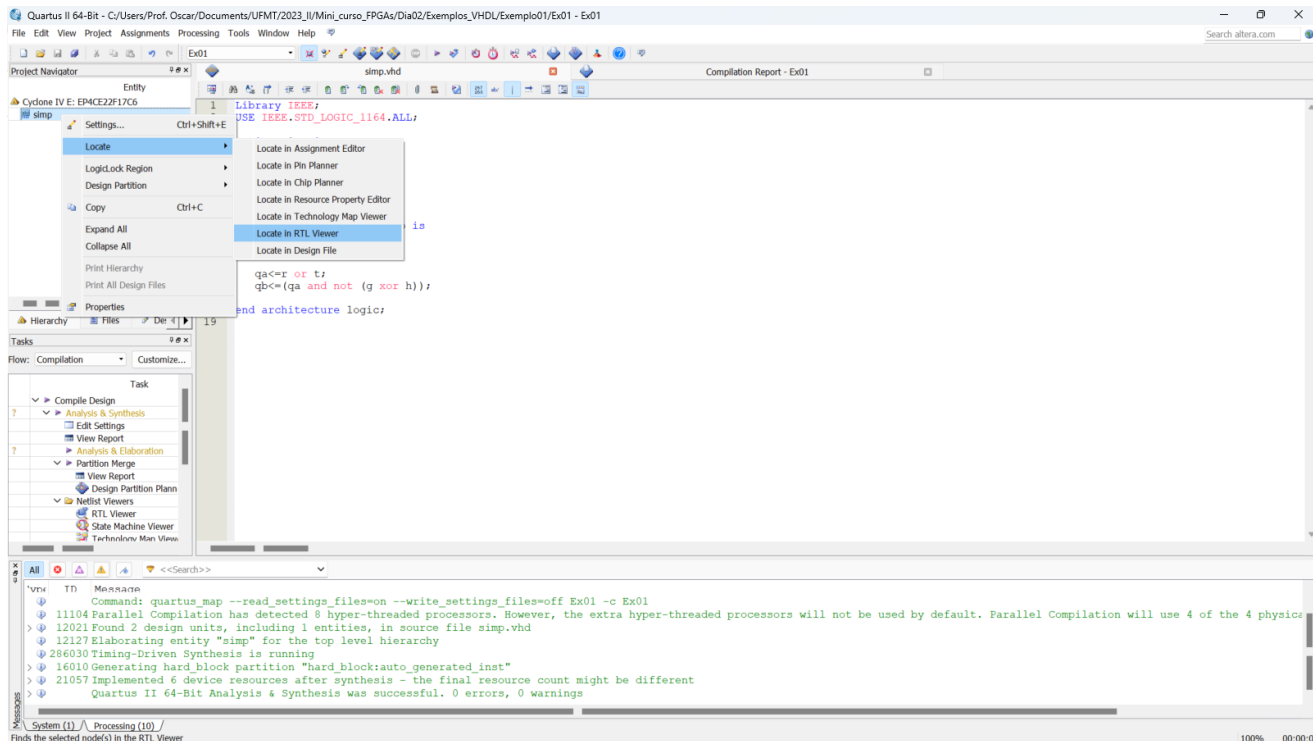
Sinais do waveform

Configuração do arquivo RTL

● Passo 1: click direito no circuito TOP



● Passo 1: Locate RTL Viewer



RTL viewer

Objetivos

RTL viewer

Testbench

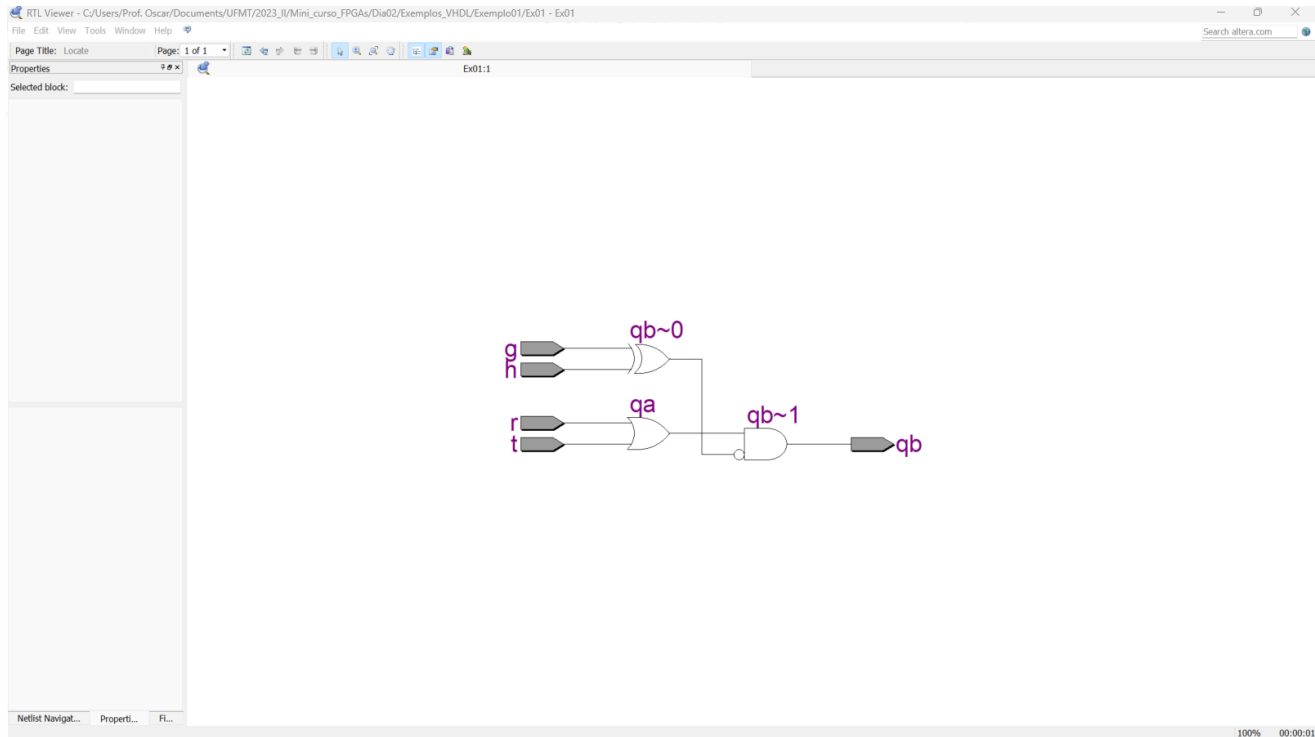
Configuração do Quartus

ModelSim

Zoom do waveform

Simais do waveform

Configuração do arquivo RTL



Arquitetura a ser verificada

Objetivos

RTL viewer

Testbench

Configuração do Quartus

ModelSim

Zoom do waveform

Sinais do waveform

Configuração do arquivo RTL

The screenshot displays the Quartus II 64-bit IDE interface. The main window shows the RTL viewer for a VHDL file named 'simp.vhd'. The code in the editor is as follows:

```

1  Library IEEE;
2  USE IEEE.STD_LOGIC_1164.ALL;
3
4  Entity simp is
5  Port(
6    r,t,g,h: in std_logic;
7    qb:      out std_logic
8  );
9  End entity;
10
11 Architecture logic of simp is
12   Signal qa: std_logic;
13 Begin
14
15   qa<=r or t;
16   qb<=(qa and not (g xor h));
17
18 End architecture logic;
19

```

The Project Navigator on the left shows the hierarchy of the project, including the 'Entity' and 'simp' components. The Messages window at the bottom displays the following message:

```

> 21057 Implemented 6 device resources after synthesis - the final resource count might be different
> Quartus II 64-Bit Analysis & Synthesis was successful. 0 errors, 0 warnings

```


Arquitetura a ser verificada

Resultado da compilação

Quartus II 64-Bit - C:/Users/Prof. Oscar/Documents/UFMT/2023_IJ/Mini_curso_FPGAs/Dia02/Exemplos_VHDL/Testbench/Ex01/Ex01 - Ex01

File Edit View Project Assignments Processing Tools Window Help

Search altera.com

tb_simp.vhd simp.vhd Compilation Report - Ex01

Entity
Cyclone IV E: EP4CE22F17C6
simp

Table of Contents
Flow Summary
Flow Settings
Flow Non-Default Global Settings
Flow Elapsed Time
Flow OS Summary
Flow Log
Analysis & Synthesis
Flow Messages
Flow Suppressed Messages

Flow Summary
Flow Status Successful - Tue Mar 19 22:13:28 2024
Quartus II 64-Bit Version 13.0.1 Build 232 06/12/2013 SP 1 SJ Web Edition
Revision Name Ex01
Top-level Entity Name simp
Family Cyclone IV E
Device EP4CE22F17C6
Timing Models Final
Total logic elements 1
Total combinational functions 1
Dedicated logic registers 0
Total registers 0
Total pins 5
Total virtual pins 0
Total memory bits 0
Embedded Multiplier 9-bit elements 0
Total PLLs 0

Task
Compile Design
Analysis & Synthesis
Edit Settings
View Report
Analysis & Elaboration
Partition Merge
View Report
Design Partition Plann
Netlist Viewers

Messages
System (1) / Processing (10) /

21057 Implemented 6 device resources after synthesis - the final resource count might be different
Quartus II 64-Bit Analysis & Synthesis was successful. 0 errors, 0 warnings

Conteúdo

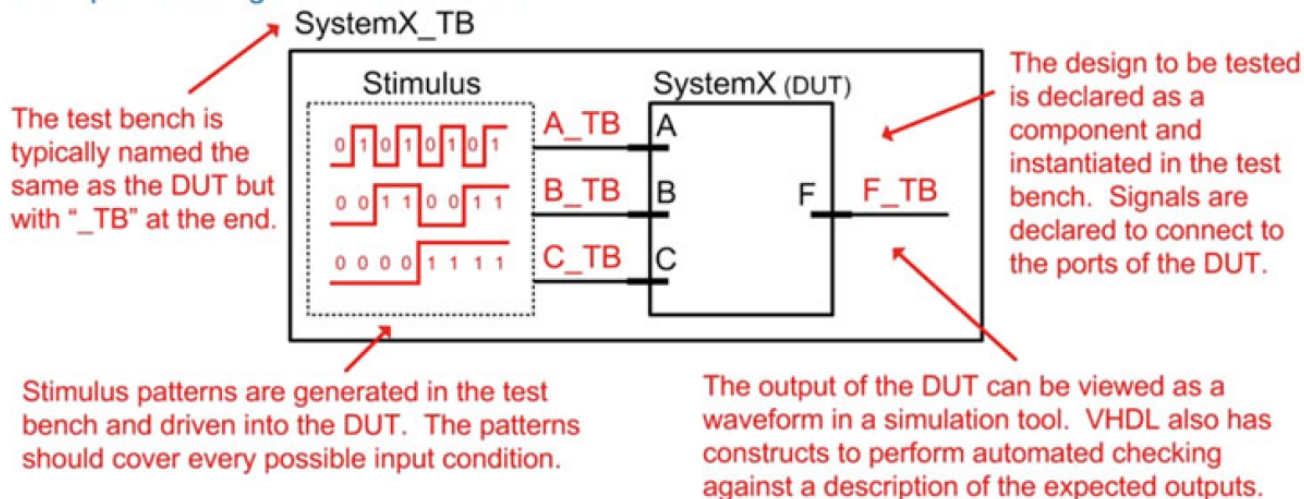
- 1 Objetivos
- 2 RTL viewer
- 3 Testbench
 - Configuração do Quartus
- 4 ModelSim
 - Zoom do waveform
 - Sinais do waveform
 - Configuração do arquivo RTL

Verificação comportamental

Testbench

A verificação funcional de arquiteturas VHDL é realizada por meio de simulação utilizando um banco de testes. Um banco de testes é um sistema VHDL que instancia o sistema a ser testado como um componente e então gera os padrões de entrada e observa as saídas. O sistema em teste é frequentemente chamado de dispositivo em teste (DUT) ou unidade em teste (UUT). Bancos de testes são utilizados apenas para simulação, então podemos utilizar técnicas de modelagem abstrata que não são sintetizáveis para gerar os padrões de estímulo.

Example: Creating a VHDL Test Bench



Testbench em VHDL

Objetivos

RTL viewer

Testbench

Configuração do Quartus

ModelSim

Zoom do waveform

Sinais do waveform

Configuração do arquivo RTL

```

entity SystemX_TB is
end entity;

architecture SystemX_TB_arch of SystemX_TB is

    component SystemX                                -- Component Declaration
        port (A, B, C : in bit;
              F      : out bit);
    end component;

    signal  A_TB, B_TB, C_TB : bit;    -- Signal Declaration
    signal  F_TB             : bit;

begin

    DUT1 : SystemX port map (A => A_TB, -- DUT Instantiation
                           B => B_TB,
                           C => C_TB,
                           F => F_TB);

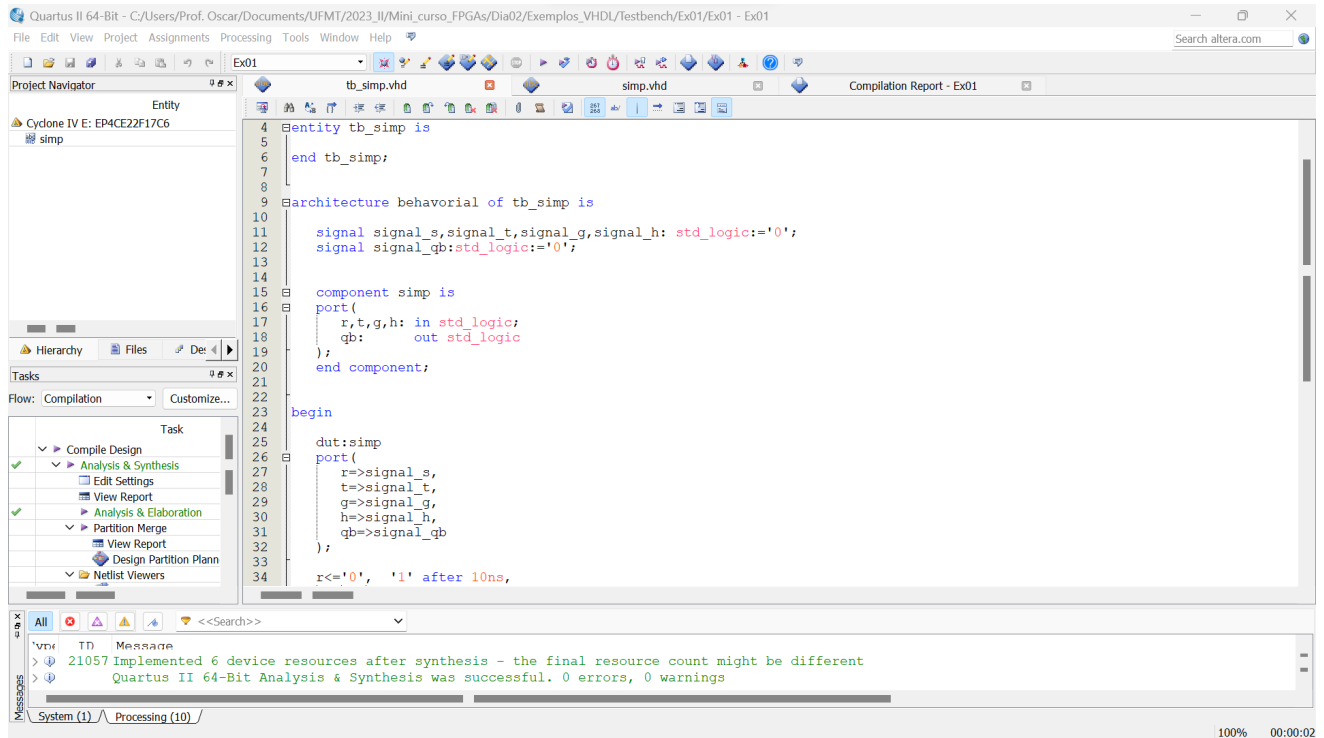
    -- Stimulus Generation
    STIMULUS : process
    begin
        A_TB <= '0'; B_TB <= '0'; C_TB <= '0'; wait for t_wait;
        A_TB <= '0'; B_TB <= '0'; C_TB <= '1'; wait for t_wait;
        A_TB <= '0'; B_TB <= '1'; C_TB <= '0'; wait for t_wait;
        A_TB <= '0'; B_TB <= '1'; C_TB <= '1'; wait for t_wait;
        A_TB <= '1'; B_TB <= '0'; C_TB <= '0'; wait for t_wait;
        A_TB <= '1'; B_TB <= '0'; C_TB <= '1'; wait for t_wait;
        A_TB <= '1'; B_TB <= '1'; C_TB <= '0'; wait for t_wait;
        A_TB <= '1'; B_TB <= '1'; C_TB <= '1'; wait for t_wait;
    end process;

end architecture;

```

Configuração do Quartus

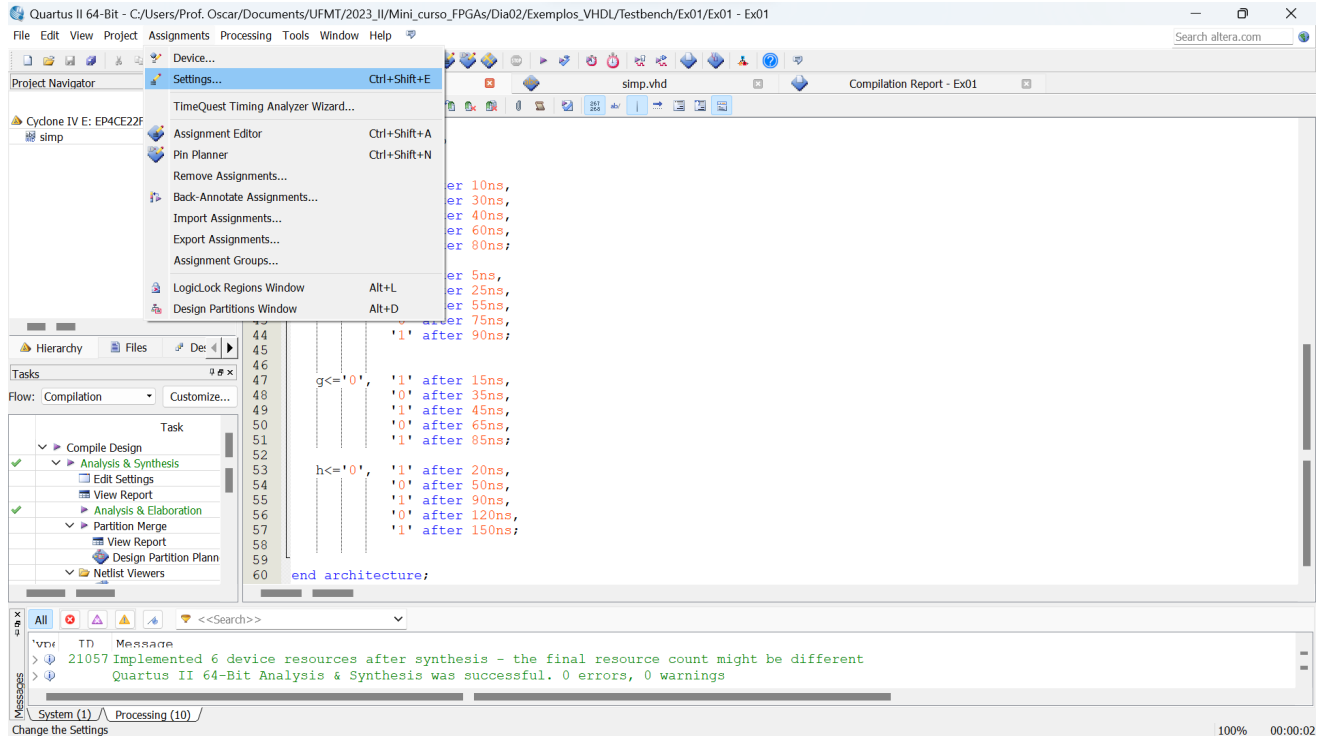
Arquivo do testbench



Configuração do Quartus

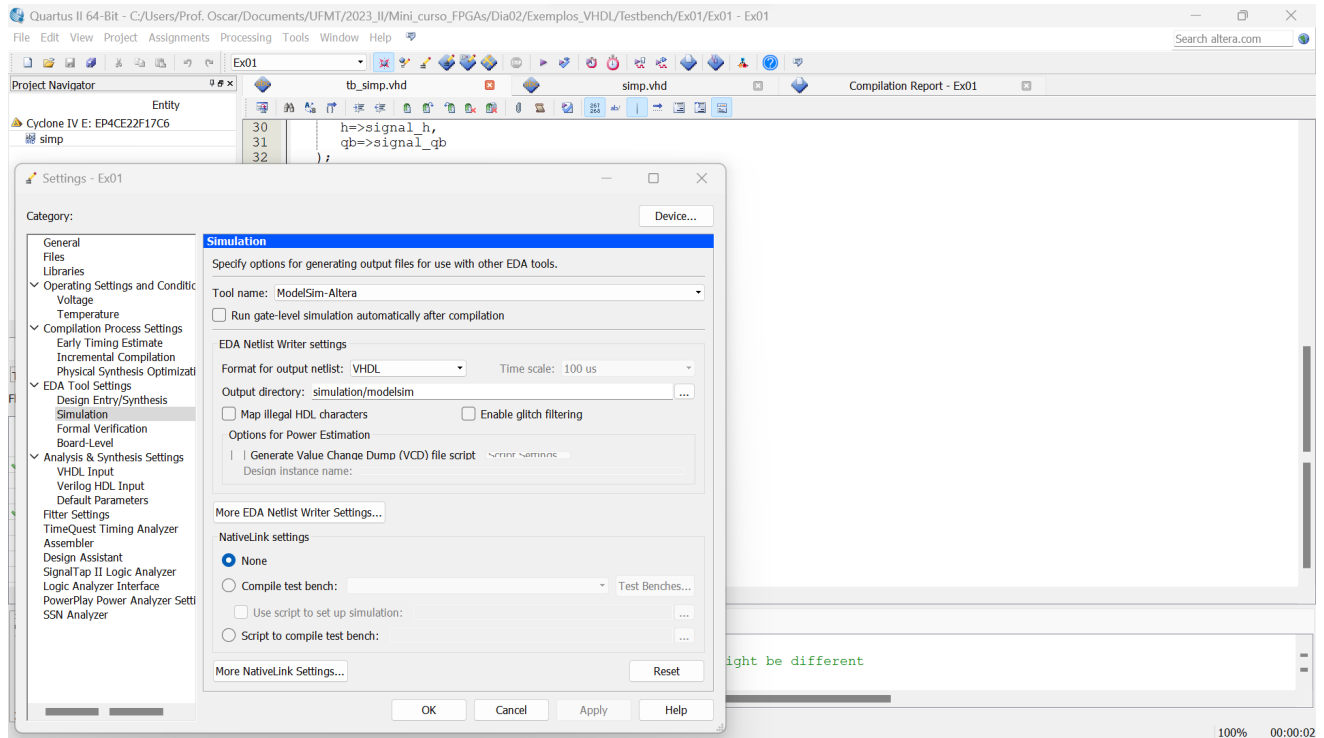
Configuração do testbench

- Passo 1: Assignments
- Passo 2: Settings



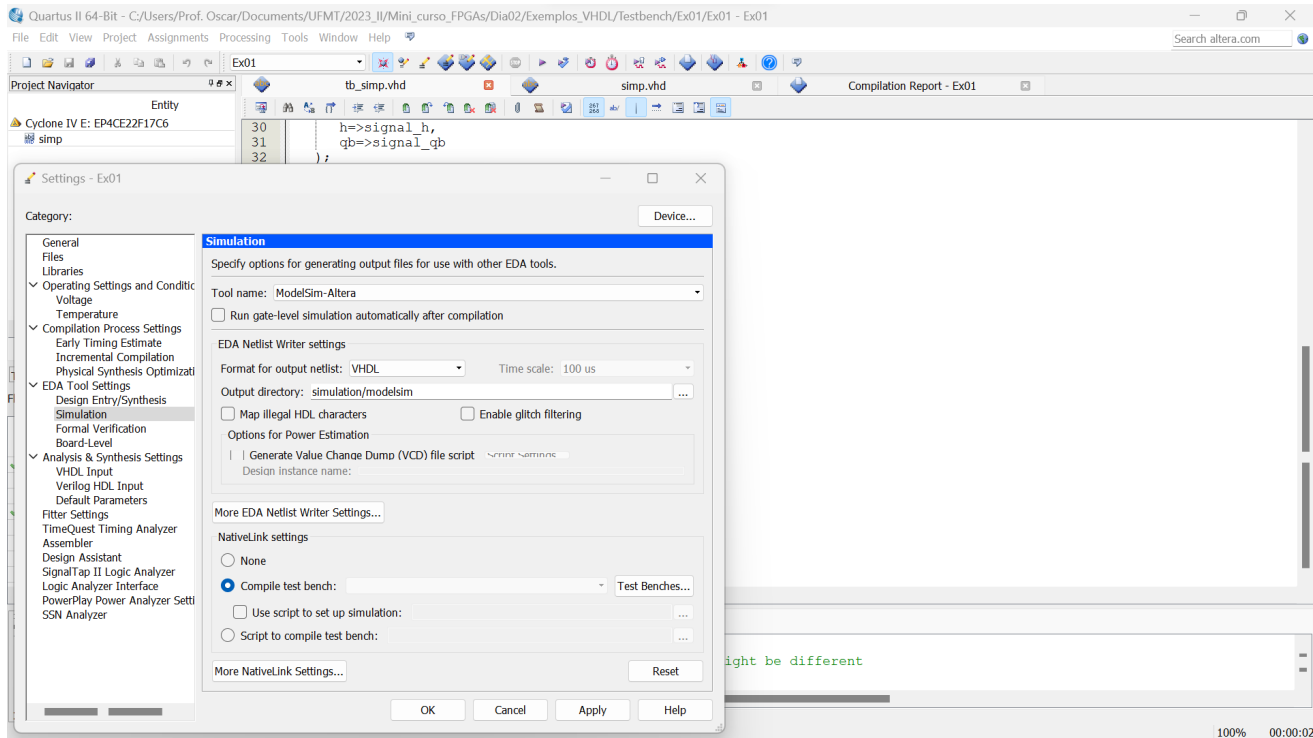
Configuração do testbench

● Passo 1: Simulation



Configuração do testbench

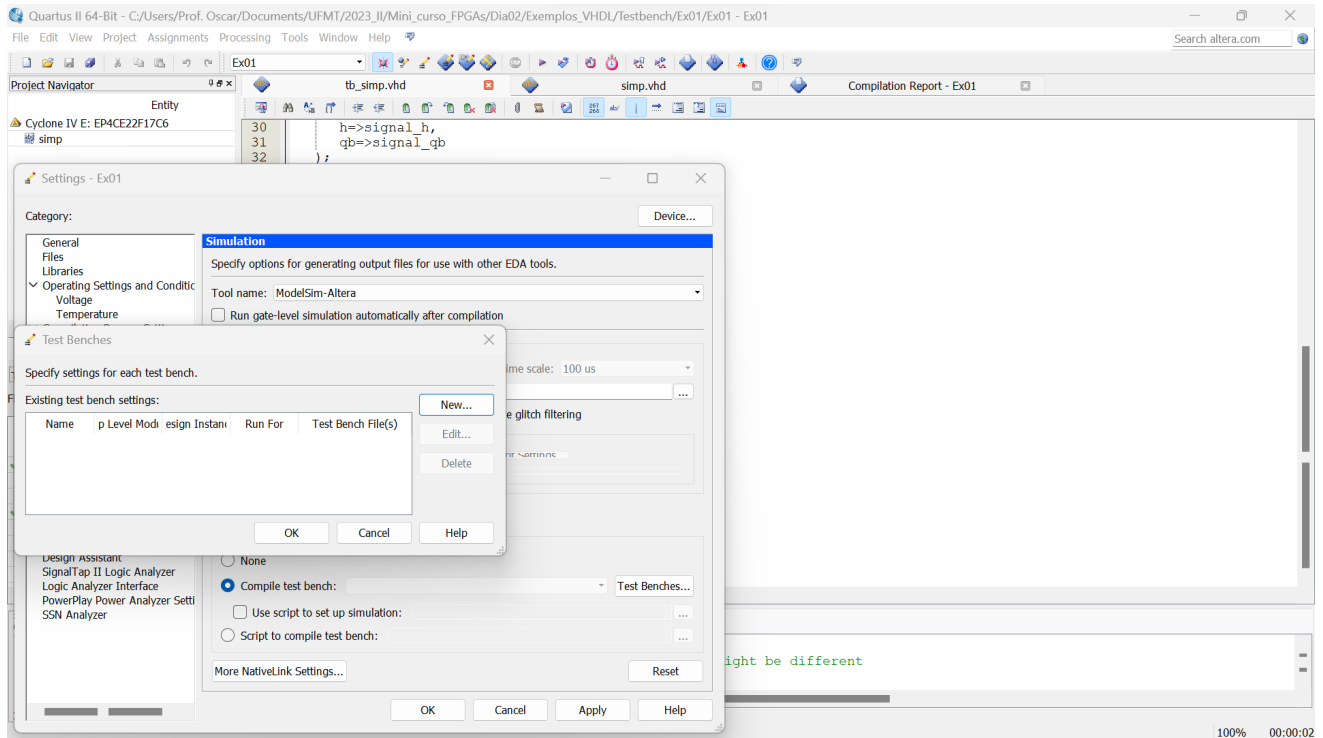
● Passo 1: Compile testbench



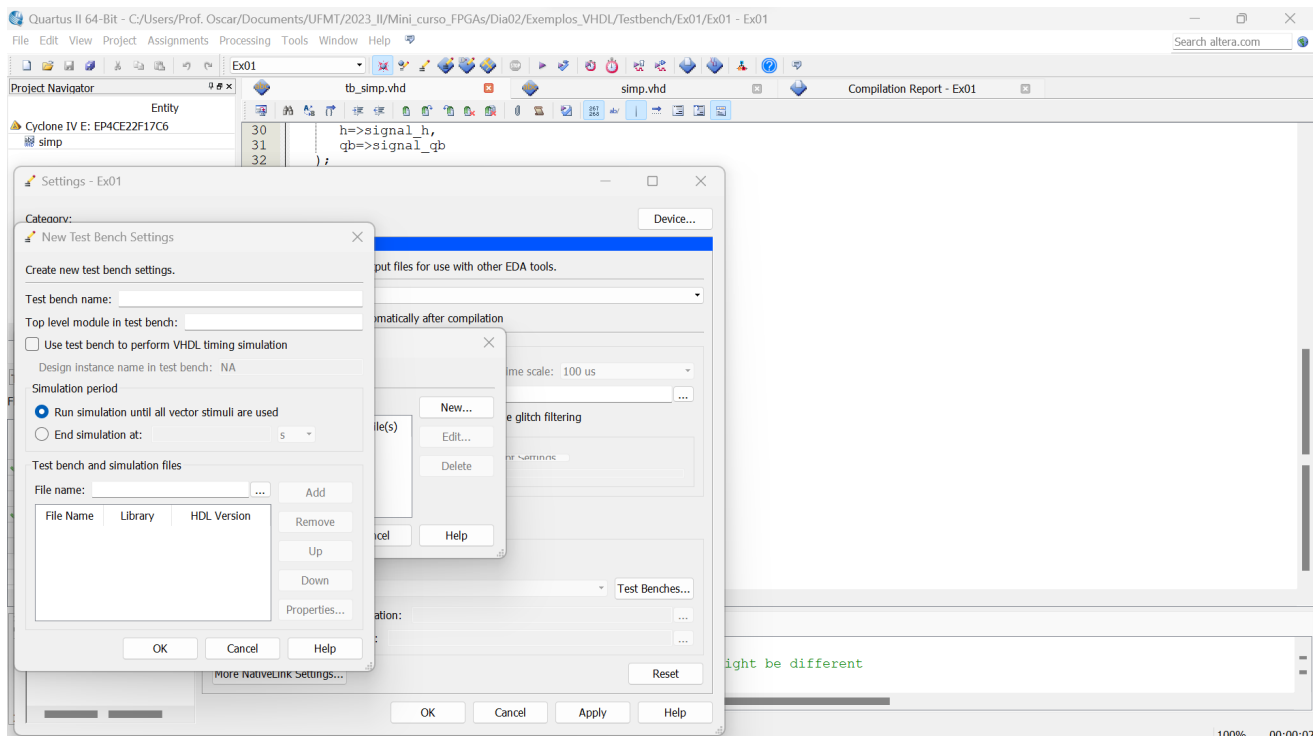
Configuração do Quartus

Configuração do testbench

● Passo 1: New...



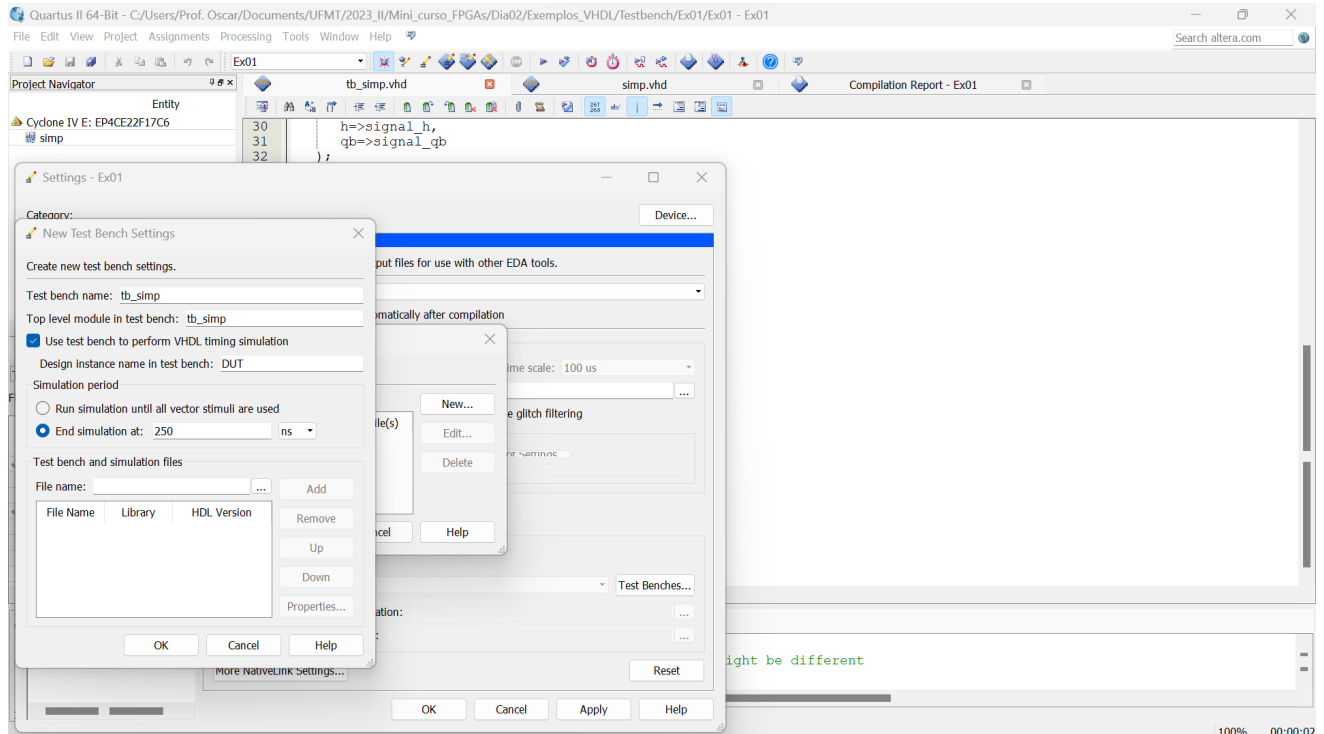
Configuração do testbench



Configuração do Quartus

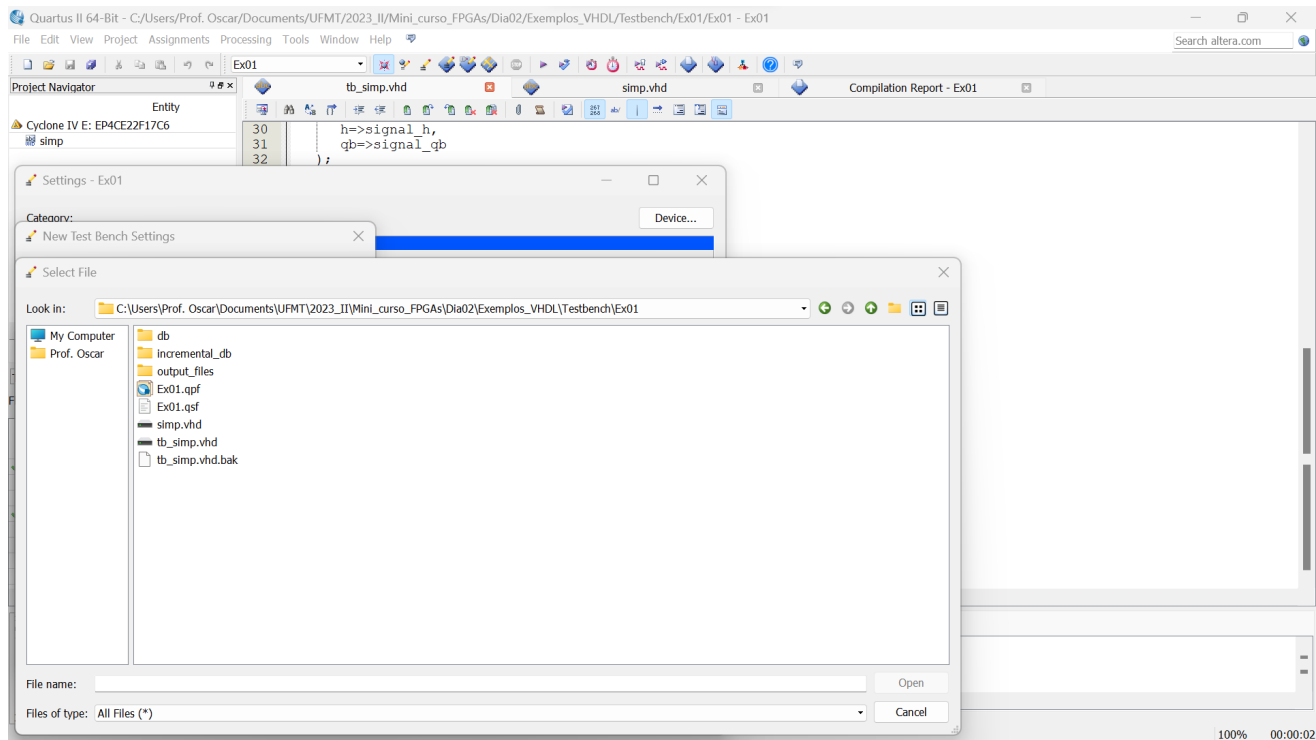
Configuração do testbench

- **Passo 1:** tb_simp
- **Passo 2:** Use test bench to perform VHDK timing simulation
- **Passo 3:** DUT
- **Passo 4:** End simulation 200 ns
- **Passo 5:** ...



Configuração do Quartus

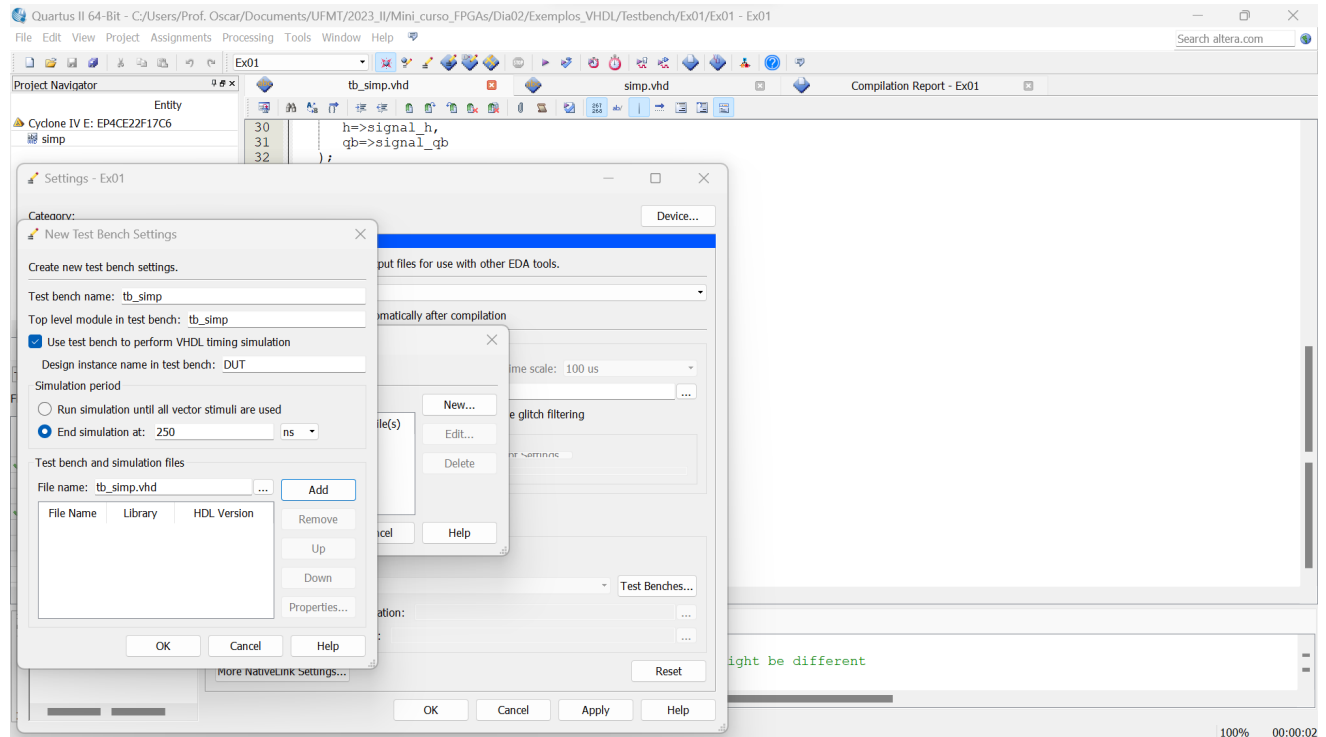
Configuração do testbench



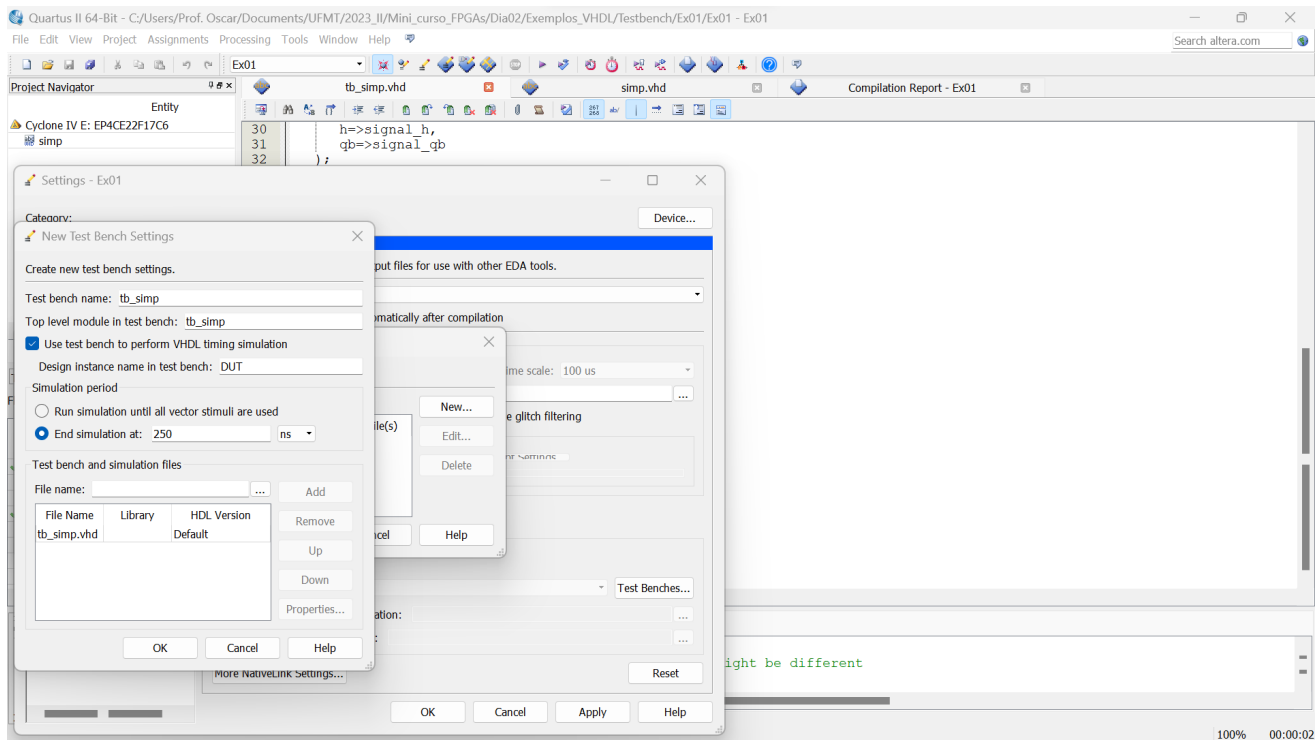
Configuração do Quartus

Configuração do testbench

- Passo 1: add
- Passo 2: OK



Configuração do testbench



Configuração do Quartus

Simulação comportamental

Quartus II 64-Bit - C:\Users\Prof. Oscar\Documents\UFMT\2023_II\Mini_curso_FPGAs\Dia02\Exemplos_VHDL\Testbench\Ex01\Ex01 - Ex01

File Edit View Project Assignments Processing Tools Window Help

Search altera.com

Project Navigator Ex01

Entity

Cyclone IV E: EP4CE22F17C6

simp

tb_simp.vhd

```

30 h=>signal_h,
31 qb=>signal_qb
32 );
33
34 r<='0', '1' after 10ns,
35 '0' after 30ns,
36 '1' after 40ns,
37 '0' after 60ns,
38 '1' after 80ns;
39
40 t<='0', '1' after 5ns,
41 '0' after 25ns,
42 '1' after 55ns,
43 '0' after 75ns,
44 '1' after 90ns;
45
46
47 g<='0', '1' after 15ns,
48 '0' after 35ns,
49 '1' after 45ns,
50 '0' after 65ns,
51 '1' after 85ns;
52
53 h<='0', '1' after 20ns,
54 '0' after 50ns,
55 '1' after 90ns,
56 '0' after 120ns,
57 '1' after 150ns;
58
59
60 end architecture;

```

Task

Flow: Compilation Customize...

Task

- Compile Design
- Analysis & Synthesis
 - Edit Settings
 - View Report
- Analysis & Elaboration
- Partition Merge
 - View Report
- Design Partition Plan
- Netlist Viewers

Messages

System (1) / Processing (10) /

Runs the specified RTL simulation tool

100% 00:00:02

Message

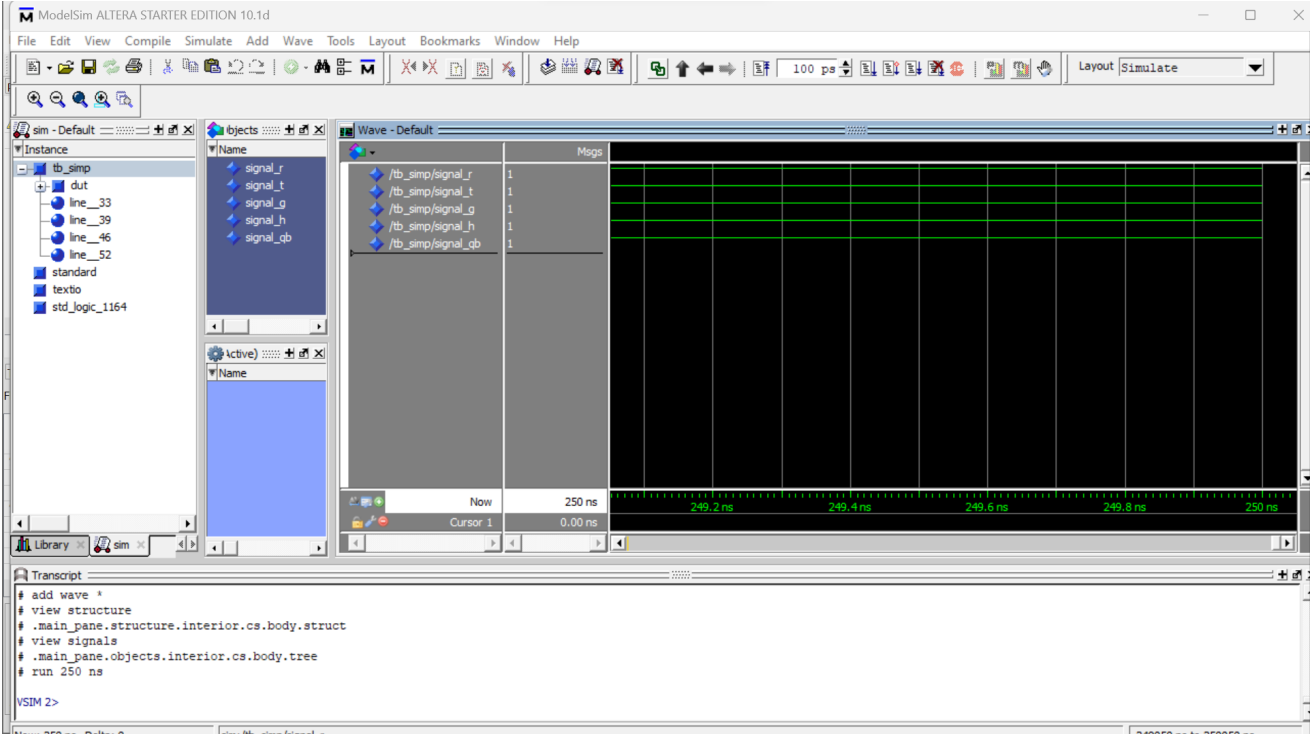
21057 Implemented 6 device resources after synthesis - the final resource count might be different

Quartus II 64-Bit Analysis & Synthesis was successful. 0 errors, 0 warnings

Conteúdo

- 1 Objetivos
- 2 RTL viewer
- 3 Testbench
 - Configuração do Quartus
- 4 ModelSim
 - Zoom do waveform
 - Sinais do waveform
 - Configuração do arquivo RTL

- Objetivos
- RTL viewer
- Testbench
- Configuração do Quartus
- ModelSim
- Zoom do waveform
- Sinais do waveform
- Configuração do arquivo RTL



Objetivos

RTL viewer

Testbench

Configuração do Quartus

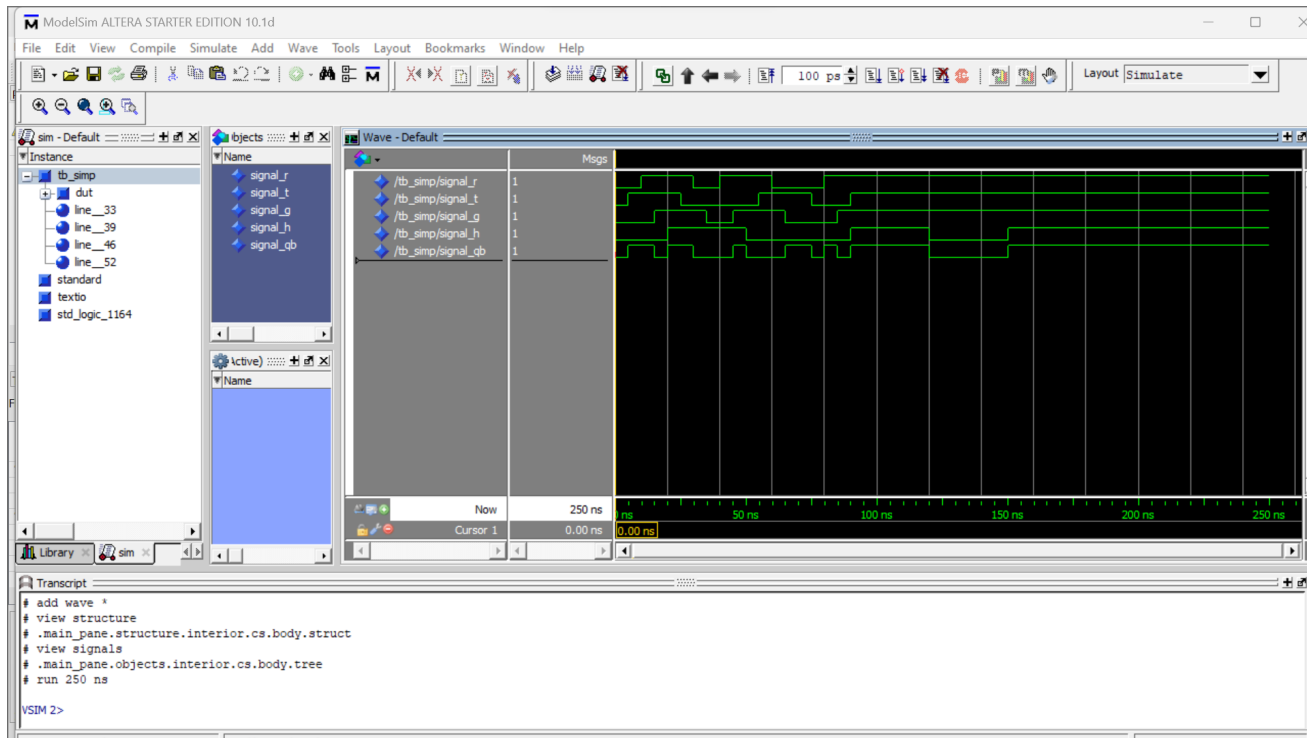
ModelSim

Zoom do waveform

Sinais do waveform

Configuração do arquivo RTL

Zoom do waveform



Objetivos

RTL viewer

Testbench

Configuração do Quartus

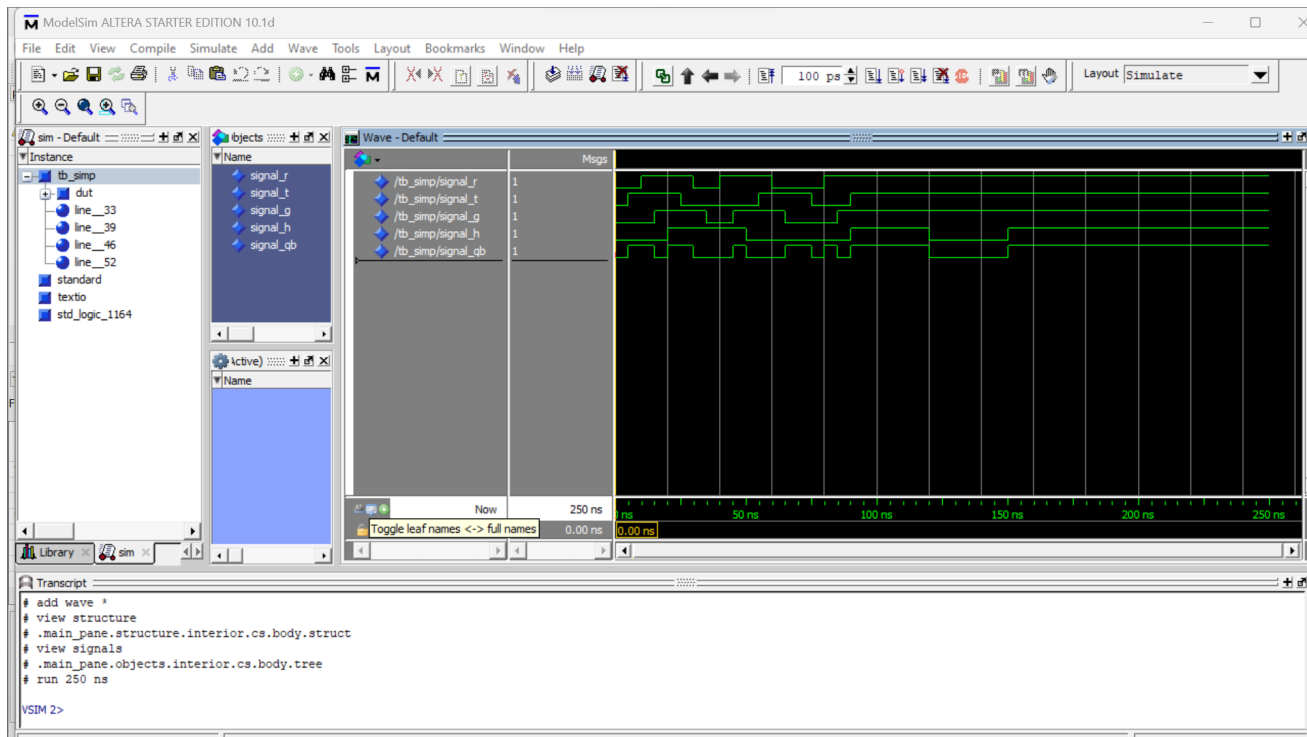
ModelSim

Zoom do waveform

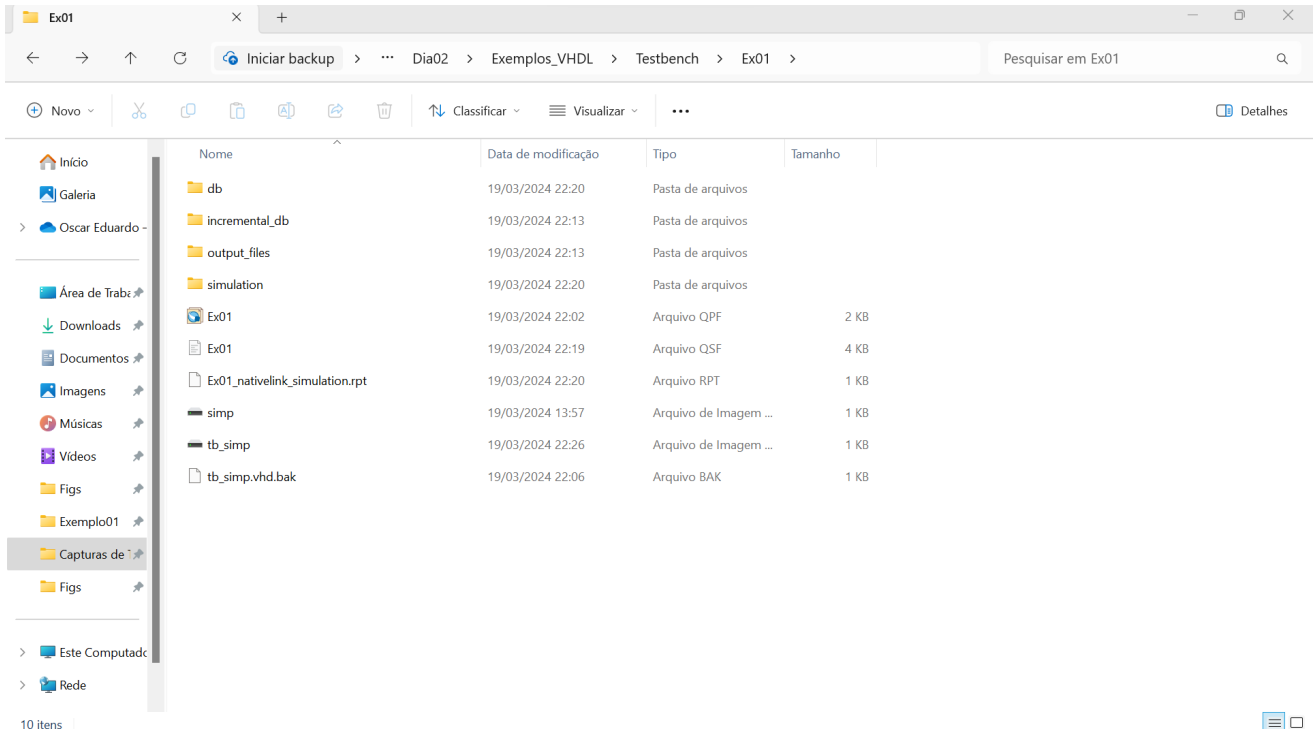
Sinais do waveform

Configuração do arquivo RTL

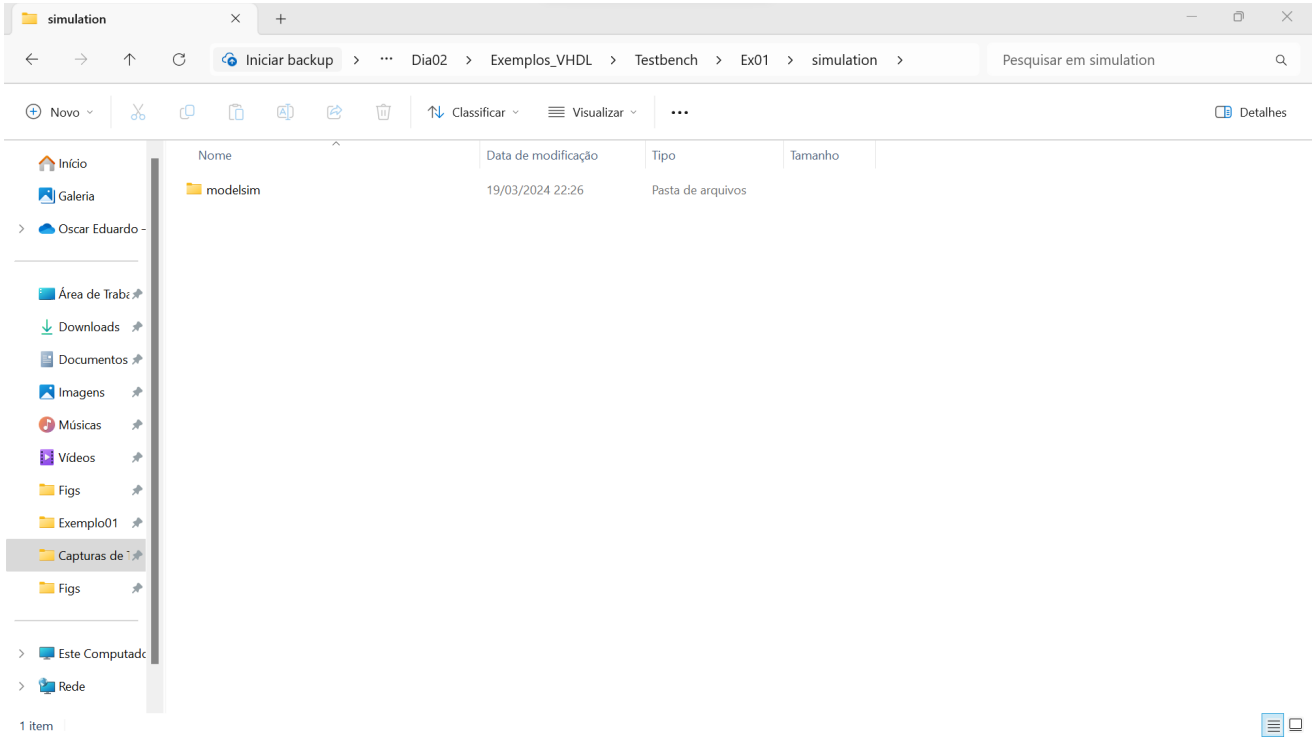
Reduzir o nome dos sinais no waveform



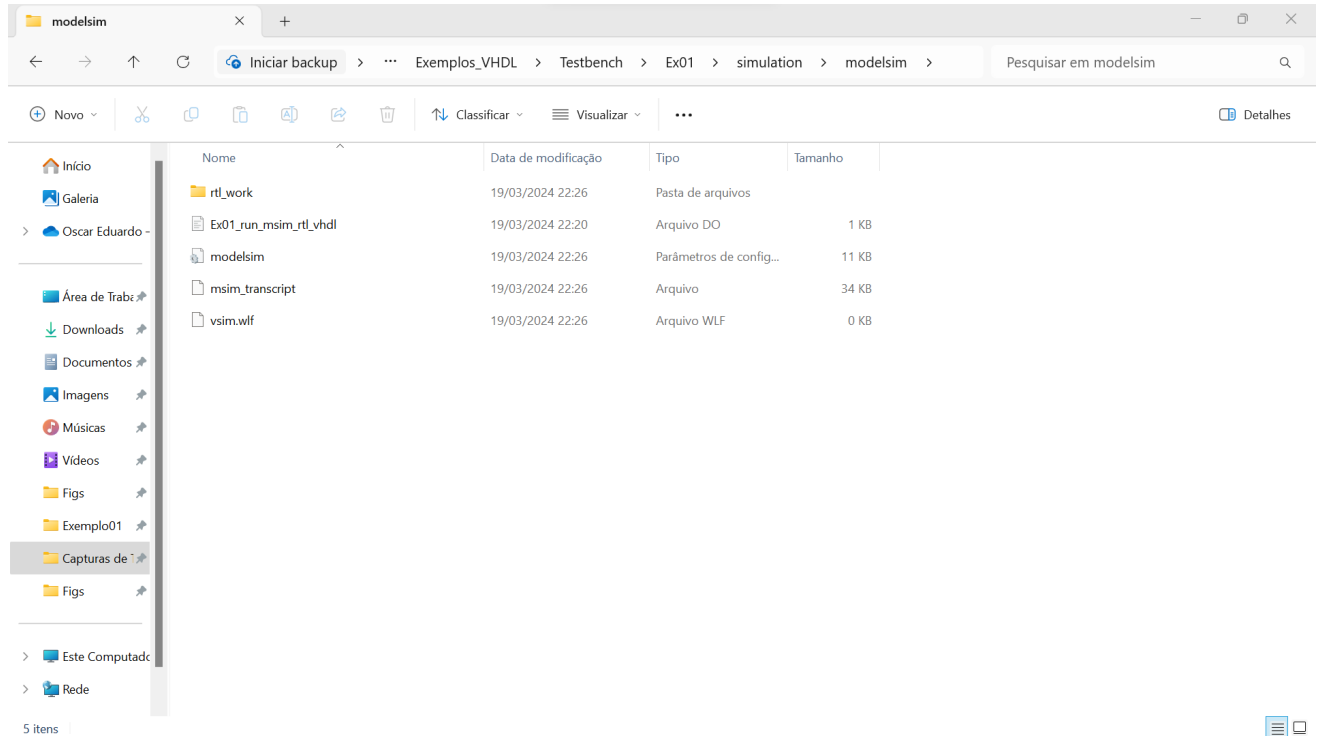
Configuração do arquivo de simulação



Configuração do arquivo de simulação



Configuração do arquivo de simulação



Abrir o arquivo rtl e adicionar a linha do wave.do

```
1  transcript on
2  ✓ if {[file exists rtl_work]} {
3    |   vdel -lib rtl_work -all
4    | }
5    vlib rtl_work
6    vmap work rtl_work
7
8    vcom -93 -work work {C:/Users/Prof. Oscar/Documents/UFMT/2023_II/Mini_curso_FPGAs/Dia02/Exemplos_VHDL/Testbench/Ex01/simp.vhd}
9
10   vcom -93 -work work {C:/Users/Prof. Oscar/Documents/UFMT/2023_II/Mini_curso_FPGAs/Dia02/Exemplos_VHDL/Testbench/Ex01/tb_simp.vhd}
11
12   vsim -t ips -L altera -L lpm -L sgate -L altera_mf -L altera_lnsim -L cycloneive -L rtl_work -L work -voptargs="+acc" tb_simp
13
14   do wave.do
15   view structure
16   view signals
17   run 250 ns
18
```